UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/691,836	10/23/2003	Ivan Leichtling	MS1-1784US	6661

22801 7590 06/15/2006

LEE & HAYES PLLC
421 W RIVERSIDE AVENUE SUITE 500
SPOKANE, WA 99201

EXAMINER	
WOODS, ERIC V	
ART UNIT	**PAPER NUMBER**
2628	

DATE MAILED: 06/15/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)
	10/691,836	LEICHTLING, IVAN
	Examiner	Art Unit
	Eric Woods	2628

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *03 February 2006*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-27* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-27* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

Response to Arguments

Applicant's arguments, see Remarks pages 1-14, filed 3 February 2006, with respect to the rejection(s) of claim(s) 1-27 under various statutes have been fully considered and are persuasive in view of applicant's amendments only.

The rejections of claims 22-23 under 35 USC 101 stands withdrawn in view of applicant's amendments to the claims to place them within one of the four statutory categories of subject matter.

The rejections of claims 1-27 under 35 USC 112, second paragraph, as indefinite for failing to adequately define the term 'synchronously' in the context of the claims stands withdrawn in view of applicant's clarification of the term in the amended versions of the independent claims.

The rejections of claims 1-27 under 35 USC 103(a) under various combinations of references stand withdrawn in view of applicant's amendments, which have clearly changed claim scope.

Therefore, applicant's arguments – which are entirely directed at the amended claims rather than the rejected ones – are found to be moot, since the grounds of rejection they are directed against stand withdrawn.

However, upon further consideration, a new ground(s) of rejection for claims 1-27 is made under 35 USC 103(a) in view of the same references as below.

It is noted that examiner strongly disagrees with applicant's contention in the Arguments (page 4, lines 18-24) that a command issued every 'x' milliseconds, which

directs a node agent to determine changes to an associated desktop environment, does **not** suggest synchronization.

When dealing with computer systems over a network, there will inherently be some kind of delay. Computers do not operate instantaneously and require certain, finite intervals of time to complete operations. Indeed, the task of determining the change of status of a window in a windowed operating system would take upwards of a hundred microseconds even on the fastest currently available workstations. However, the more relevant inquiry is to the effect **network latency** has on the response time. More precisely, computers take time to transmit information. Since the instant application is directed to **remote** control of computers and transmission of information over the Internet or other network, that is the controlling factor. Typically, even over dedicated links, such data transmission (e.g. framing, queuing, division into packet, transmission, reception, and the other relevant processing) as well as transmission delays will result in delay factors that exceed several milliseconds. As such, a computer cannot refresh a remote desktop at rates upwards of 30-60Hz in a practical manner (indeed, typical full-motion video for video conferencing refreshes at 30fps or 30Hz. The human brain does not notice motion blur at speeds in excess of 30-40fps in any case. Finally, currently available monitor hardware does not allow for a refresh rate greater than 60Hz. It is noted that these rates (30-60Hz) correspond to intervals of 16ms-33ms. Therefore, that is consistent with network transmission times – from end-to-end – that are typically in the range of 5-10ms or higher. Current hardware and

networking simply does not physically allow rates that are higher than this in the first place.

Next, computers operate in discrete steps rather than continuously. They perform in a clocked manner, where this consists of a discrete series of steps. Therefore, computers execute certain instructions over a certain time period. Repeating a set of instructions (e.g. determining changes in node status) will take place at discrete time intervals even in the case of 'synchronous' data gathering.

Therefore, as noted and discussed above, it is entirely credible that a reference that teaches checking a node every "X" milliseconds would teach synchronicity as described above and as examiner has discussed in the last Office Action.

Next, on page 4, lines 1-17, applicant argues that Panasyuk does not teach 'synchronously gathering' region and graphics data are inapposite.

Firstly, applicant defines 'graphical data' or 'graphics data' to be (instant specification, page 2, first 5 lines of second paragraph): "...information that makes up the visual content of the desktop or a region of the desktop, and sends this graphics data back to the client. The graphics data may describe text, image content, and/or UI features, such as scroll bars and clickable buttons generated by the server ..."

Applicant defines "region data" in the bottom of the paragraph, where applicant states: "Thus, to accurately display a region from the server's desktop on its own display, the client must be informed of the region's graphics data, **shape, and position** (the **latter two being 'region data'**)..."(emphasis added).

In response to applicant's argument that the references fail to show certain features of applicant's invention, it is noted that the features upon which applicant relies (i.e., synchronicity of the region and graphics data on the client, with some implication that the client updates or changes its window after changes in the server) are not recited in the rejected claim(s). Although the claims are interpreted in light of the specification, limitations from the specification are not read into the claims. See *In re Van Geuns*, 988 F.2d 1181, 26 USPQ2d 1057 (Fed. Cir. 1993). Specifically, the claim(s) only require that the region and graphics data be gathered synchronously and transmitted in such a way as their association is maintained. There is nothing in the claim that requires that the client do anything with the received data.

Note that in the specification of the instant application, it is stated on page 3 that 'Asynchronous collection of region data means, for one thing, that the region data collection operates independently of the graphics data collection, i.e., without reference to matching a segment of the graphics data with a segment of the region data'). However, it is respectfully pointed out that the Panasyuk reference clearly teaches that the system contains a local node, a local agent, a first remote node, and a first remote agent (1:59-65). The remote node transmits messages indicative of changes in the first remote desktop environment. The local node receives the messages and commands the system to modify a local representation of the window on the server desktop. Note that in 2:30-40 it is explicitly specified that nodes exchange window position, window size, and bordering information (e.g. 'region data') over the first virtual channel. They

exchange graphical information over the second virtual channel (e.g. 'graphics data'). The two channels can be the same.

The system obtains region data at certain intervals by various means, but can do so via the Enum_Windows command in Microsoft Windows™ based operating systems (3:60-4:12) at regular intervals (e.g. 50ms, etc., where this is user-selectable and the requirements are that 1) the period must allow the agent to rapidly determine when changes to its associated desktop environment have occurred and 2) it must be done without placing a significant computational burden on the node). Obviously, the faster the refresh interval, the more accurate it will be.

In another embodiment, the system watches the message queues for the operating system to determine changes. In both cases, changes to region data are detected in this manner.

Further, in one embodiment, the system of Panasyuk creates and maintains windows on the client (local) desktop matching those of the server in position, size, etc (see 6:60-7:1). The system of Panasyuk additional states: "...In some embodiments, window elements are transmitted as bitmaps from the server node 20..." This clearly states that window elements (e.g. graphical content) are transmitted as well as the other information.

However, Panasyuk would suggest using synchronization since (9:55-10:10) the system tracks the order of windows displayed. As such, when the window order changes, windows can become obscured or 'clipped' as a result on the server desktop, and the system then determines whether or not to display the graphical data based on

whether or not it is obscured by applying clipping functions. One embodiment also determines whether or not such data is obscured at the time it is received. If it is, it is ignored. Turning back to 8:20-30, it is found that Panasyuk teaches that it is desirable to reduce network traffic by comparing the current window to the last window to determine whether or not any changes have occurred; if none have, no transmission is needed.

As such, transmitting graphics data that would only generate extraneous network traffic is not desirable. In order to make such a determination, a clipping function must be applied, where it can be applied at the time the graphical information is received, but would better be applied before it is sent, as in 8:20-30, to avoid unnecessary network traffic. Note further that such information for 'seamless windowing mode' can be transmitted on the same channel (2:39-41); that is, graphical and region data can be transmitted on the same channel. Their synchronization would be ideal for purposes of minimizing the amount of graphics data needed to be transmitted, as described above, since any changes in window, size, focus, position, etc, will necessarily result in some change in the graphics data, which would therefore require transmission of such information to update the local client as per the above scenario, since it is known that graphical content can be sent with such region data (e.g. graphical bitmaps as above), which would clearly **suggest** or **imply** synchronization.

Applicant then argues that (page 4, line 22- page 5, line 6) there are two agents on each server, etc. However, that is not required – indeed, the first embodiment disclosed only has one agent on the server, and one on the client. Additionally, it is

irrelevant if the system **can** accommodate multiple desktops, as the embodiments exactly mirror changes in the server desktop on the local desktop, and operate in full screen mode by default anyway (7:1-5), and the 'seamless windowing' mode can do so as well.

In response to applicant's argument that the references fail to show certain features of applicant's invention, it is noted that the features upon which applicant relies – region data and graphics data gathered across a **single desktop** -- (i.e., "there is no indication in Spencer that region data and graphics data of a single desktop display are gathered synchronously so as to maintain an association of the region data and graphics data," (Remarks page 5, lines 11-17)) are not recited in the rejected claim(s). Although the claims are interpreted in light of the specification, limitations from the specification are not read into the claims. See *In re Van Geuns*, 988 F.2d 1181, 26 USPQ2d 1057 (Fed. Cir. 1993).

Additionally, examiner has explained above how the first discussed embodiment of Panasyuk teaches the system with a single remote node and single local node, which therefore only gathers region data and graphics data for a single desktop.

Examiner respectfully points out there is simply no recitation in the claim of what the client does with the received data, there is no recitation that the client cannot send information back to the remote server desktop, and the like. There is finally nothing that says only one server is or is not required, etc.

The only question is whether or not there is sufficient teaching in Panasyuk to synchronize the data streams. While examiner believes that there is at least a

suggestion within Panasyuk to do so, for purposes of expediting prosecution, a second reference is brought in.

The Spencer reference is used to provide the additional motivation and teaching for synchronizing region and graphics data.

Applicant's arguments with respect to claim 15 are moot because examiner has changed base reference.

Applicant's arguments with respect to claim 20 are not complete; examiner has clarified the means below. Further, applicant characterizes the Panasyuk reference as not sending graphics data from the server to the client, which is not found to be accurate, since the local and remote (server) computers have agents that send data between them, and the remote agent transmits messages to the client agent, as found in 3:58-6:45, in the discussions of packet dynamics and the like.

Claim Objections

Claims 1, 22, and 24 are objected to because the method taught requires a computer and the claims do not recite a computer. Applicant needs to amend the preamble of these claims to include the words 'computer-implemented' before 'method'.

Definitions

The applicant in the claims uses the term "synchronously". This term does not have a definition in the specification. Specifically, it is unknown if this term requires real-time synchronization, or if it is used in a broad manner to simply require that events on the server and the client be updated, usually as practical (e.g. in high-latency and/or

high packet-loss scenarios)(where practicality is entirely determined by the reference and the situation, such as in Panasyuk (4:1-12), where information concerning movement of the window is transmitted when it 'would not cause a significant computation burden on the node'. As such, examiner will interpret the term as broadly as possible. This was not clarified in response to the last Office Action, and it would appear to be reasonable to interpret it as above.

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

The factual inquiries set forth in *Graham* **v.** *John Deere Co.*, 383 U.S. 1, 148 USPQ 459 (1966), that are applied for establishing a background for determining obviousness under 35 U.S.C. 103(a) are summarized as follows:

1. Determining the scope and contents of the prior art.
2. Ascertaining the differences between the prior art and the claims at issue.
3. Resolving the level of ordinary skill in the pertinent art.
4. Considering objective evidence present in the application indicating obviousness or nonobviousness.

Claims 1, 8-9, 14, 19-21, and 25-26 are rejected under 35 U.S.C. 103(a) as being unpatentable over Panasyuk in view of Spencer.

As to claims 1, 20, and 25-26, (1 is method claim, 25 is computer program product claim that executes method of claim 1, and 20 is system claim implementing method of claim 1; additional limitations of system claim will be addressed in sub-clause

after main body of rejection. Same grounds of rejection are therefore applicable against all three claims)

A method, comprising: (Preamble is not given patentable weight, since it only recites a summary of the claim and/or an intended use, and the process steps and/or apparatus components are capable of standing on their own; see Rowe v. Dror, 112 F.3d 473, 42 USPQ2d 1550 (Fed. Cir. 1997), Pitney Bowes, Inc. v. Hewlett-Packard Co., 182 F.3d 1298, 1305, 51 USPQ2d 1161, 1165 (Fed. Cir. 1999), and the like.)

-Gathering region data for displaying a region of a server desktop remotely on a client display, wherein the region data describe a shape and a position of the region; (Panasyuk reference clearly teaches that the system contains a local node, a local agent, a first remote node, and a first remote agent (1:59-65). The remote node transmits messages indicative of changes in the first remote desktop environment. The local node receives the messages and commands the system to modify a local representation of the window on the server desktop. Note that in 2:30-40 it is explicitly specified that nodes exchange window position, window size, and bordering information (e.g. 'region data') over the first virtual channel. The system obtains region data at certain intervals by various means, but can do so via the Enum_Windows command in Microsoft Windows™ based operating systems (3:60-4:12) at regular intervals (e.g. 50ms, etc., where this is user-selectable and the requirements are that 1) the period must allow the agent to rapidly determine when changes to its associated desktop environment have occurred and 2) it must be done without placing a significant computational burden on the node). Obviously, the faster the refresh interval, the more

accurate it will be. In another embodiment, the system watches the message queues for the operating system to determine changes. In both cases, changes to region data are detected in this manner.)

-Gathering graphics data for the region, wherein the graphics data describe visual content of the region, and wherein the region data and the graphics data are gathered synchronously so as to maintain an association of the region data and the graphics data; and (Panasyuk creates and maintains windows on the client (local) desktop matching those of the server in position, size, etc (see 6:60-7:1). The system of Panasyuk additional states: "... In some embodiments, window elements are transmitted as bitmaps from the server node 20..." This clearly states that window elements (e.g. graphical content) are transmitted as well as the other information. Further, the system of Panasyuk always transmits graphical data; if the system is not in "seamless windowing" mode it may not be as smooth or immediately resized as otherwise, but the graphical data is still automatically transmitted, as in Figure 2)

-Sending the region data and the graphics data to a client while maintaining the association between the region data and the graphics data. (Panasyuk teaches that the region and graphics information can be transmitted on the same virtual channel (Column 2, lines 31 – 42), where such graphical data is obviously transmitted. See Panasyuk 6:60-7:1. The system of Panasyuk additional states: "... In some embodiments, window elements are transmitted as bitmaps from the server node 20..." This clearly states that window elements (e.g. graphical content) are transmitted as well as the other information. Panasyuk would suggest using synchronization since (9:55-

10:10) the system tracks the order of windows displayed. As such, when the window order changes, windows can become obscured or 'clipped' as a result on the server desktop, and the system then determines whether or not to display the graphical data based on whether or not it is obscured by applying clipping functions. One embodiment also determines whether or not such data is obscured at the time it is received. If it is, it is ignored. Turning back to 8:20-30, it is found that Panasyuk teaches that it is desirable to reduce network traffic by comparing the current window to the last window to determine whether or not any changes have occurred; if none have, no transmission is needed.)(Spencer clearly teaches synchronization verification of multiple applications across remote systems, where at least one local application window is synchronized with the at least one remote application window (Abstract, 2:45-3:7, Figs. 7C and 7G). Specifically, the system provides automated real-time feedback as to the synchronization of all windows and thusly allows synchronization between them (4:59-5:27))

Panasyuk teaches all the limitations of the instant claim except expressly teaching the use of synchronization between graphics and region data. However, it should be noted that the "association" is merely that the two forms of data are synchronized. As noted above in the Response to Arguments section (which is incorporated by reference), Panasyuk at least suggests the benefits of synchronicity if not ever specifying it directly.

Spencer teaches that applications should be synchronized across computers and that multiple applications may be so synchronized (1:25-60), and that such

synchronization is important and improves productivity (1:18-40). Panasyuk definitely teaches that the server and client have different programs and windows running on them, since they monitor the z-order and focus of windows and are concerned with clipping (3:60-4:40, 9:55-10:20). Therefore, it would be important to keep the applications synchronized between the remote server and the local client. For such synchronization to occur, it would need to be seamless. Therefore, the Spencer reference teaches that synchronizing applications in a transparent manner would be optimal.

However, Panasyuk would suggest using synchronization since (9:55-10:10) the system tracks the order of windows displayed. As such, when the window order changes, windows can become obscured or 'clipped' as a result on the server desktop, and the system then determines whether or not to display the graphical data based on whether or not it is obscured by applying clipping functions. One embodiment also determines whether or not such data is obscured at the time it is received. If it is, it is ignored. Turning back to 8:20-30, it is found that Panasyuk teaches that it is desirable to reduce network traffic by comparing the current window to the last window to determine whether or not any changes have occurred; if none have, no transmission is needed.

Clearly, it would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the system of Panasyuk such that when both the graphics data and region data are sent using one channel that such transmissions are synchronized with Spencer because Spencer provides methods for synchronizing

applications and ensuring that such synchronization is maintained in real time in a low bandwidth and transparent manner (4:59-5:27), where clearly it provides improvement over prior art window sharing (7:44-50), where clearly decreasing the amount of network throughput required (8:20-30 Panasyuk) makes it more efficient and less of a drain on the remote node (4:3-10).

As to claim 20 specifically, the means are merely the corresponding software elements of the Panasyuk reference. The 'means for producing visual content' is merely the graphic hardware of the server itself, which it inherently must possess in the system of Panasyuk. The 'means for designating' is merely the Windows interface which transmits the window information, including size, z-order, and the like, as described therein. The remote agent of Panasyuk at the server constitutes a 'means for gathering', where the synchronicity is suggested by Spencer as above, and the data is transmitted in a synchronous manner as described in the rejection to claim 1 above, which is incorporated by reference. The 'means for sending' is the network link described in the Panasyuk reference. Next, the local and remote agents send data to each other.

Claim 8 is disclosed by the invention of Panasyuk such that the region data is sequenced to precede the graphics data using rules of a remoting protocol. Column 3, lines 1 – 4, states, "Client nodes may communicate with server node 20 via any of a number of industry-standard data communications protocols including, but not limited to, TCP/IP, IPX/SPX, NetBEUI, or serial protocols." Thus, the rules of a remoting protocol

are used while the synchronized region data precedes the graphics data are sent from the server to the client.

Claims 9, 14, 19, and 21 are disclosed by the invention of Panasyuk and Spencer. The rejection to claim 1 is incorporated by reference in its entirety. Column 6 describes the sending of window information from the server to the client. Lines 63 – 66 state, "In accordance with this information, the client agent 40 creates windows with the same size/position as the server node windows on the client node desktop." Thus, the client receives the region and graphics data from the server and displays the graphics data in accordance with the region data. Finally, these claims are broader (in some cases, as in claim 19, than the claims rejected under the rejection to claim 1 above. The courts have stated on numerous occasions that the omissions of an element and its function in combination is an obvious expedient if the remaining elements perform the same function as before (see <u>In re Karlson</u> (CCPA) 136 USPQ 184 (1963)). Therefore, such a broader version is an obvious expedient. Also, the means recited in claim 21 are the client and remote agents. Further clarification will be made in Examiner's Answer at applicant's request.

As to claim 26, as discussed in the rejection to claim 1, the region data is synchronized to the graphics data when sent.

Claims 10 – 13 are rejected under 35 U.S.C. 103(a) as being unpatentable over Panasyuk in view of Spencer as applied to claims 1 above, and further in view of Fyles.

Panasyuk and Spencer et al. disclose the invention in claims 10 – 13 except wherein various methods are used to reduce the amount of information sent from the server to the client display during conditions of low bandwidth. Spencer teaches that his invention uses little bandwidth, as noted above (4:59-5:27), but otherwise does not address the problem. The invention of Fyles discloses a system for efficient workstation screen updates that involves sending display information from a local computer to a remote computer. Fyles teaches of various ways of coping with low bandwidth situations while transmitting data from one computer to another. Column 1, lines 36 – 42, states, "A major problem in achieving this simultaneity between workstations is that the connections between the computers have a limited bandwidth. This is particularly so if telephone-based ISDN lines or similar are used. One way of coping with this is to use data compression, to reduce the amount of data that must be transmitted." Column 2, lines 17 – 27, states, "In a preferred embodiment each identified portion of the screen is represented by a rectangle, and it is then the contents of this rectangle that is transmitted to the other computers in the network. The use of a rectangle is computationally very simple, and turns out to correspond to a large majority of updates. In a few cases the update has a more complicated shape, so that possibly a large proportion of the rectangle transmitted has not been updated. This could be avoided by using other shapes, perhaps based on more sophisticated calculations to determine very accurately the updated area of the screen for transmission." Thus, the system of Fyles solves the restrictions caused by low bandwidth situations by altering the shape of the area to be sent for updating to the remote computer to reduce the amount of data

that is to be transmitted. It would have been obvious to one skilled in the art at the invention was made to further modify the invention of Panasyuk in view of Spencer to include altering the area to be sent to the client computer so that less data needs to be transmitted as taught by Fyles et al. One would have been motivated to make such a modification so that in cases where the bandwidth is too low to send the graphics and region data, the region data and graphics data may be altered in order to reduce the amount of bandwidth required by the transmission. It is also inherent in the invention of Panasyuk as modified by Spencer that a local computer may only transmit as much data to a remote system as allowed by the available bandwidth between the two. During situations of low bandwidth, only a reduced amount of data can be transferred.

Claims 2-3, 5-6, and 27 are rejected under 35 U.S.C. 103(a) as being unpatentable over Panasyuk Spencer as applied to claims 1 and 25 above, and further in view of Schneider.

Panasyuk and Spencer et al. disclose the invention in claims 2, 5, and 27 except wherein the region and graphics data are synchronously gathered in a single driver. The system of Schneider controls the viewing of a display from a local computer on a target device by transmitting GDI calls. Figure 3a shows an implementation of the invention in which screen data is sampled and captured for transmission. Column 6, lines 65 – 67, and Column 7, lines 1 – 7, state, "Instead, the digitizer control application 220 periodically requests (through the device driver 210) that a whole screen of data be sampled. The digitizer control application 220 then draws the whole captured screen to

its local screen using Windows GDI calls. The remote control software application 200 captures those GDI requests and retransmits them to the controlling computer 12. The client software on the controlling computer 12 then re-executes the commands so that the screen of the controller 50 and the screen of the controlling computer 12 show the same image." Thus, the device driver and control application of Schneider et al. collect the screen data, which includes region and graphics data to be stored in the data structure of Spencer for synchronous transmission. It would have been obvious to one skilled in the art at the invention was made to further modify the invention of Panasyuk in view of Spencer so that the region and graphics data are synchronously gathered by the display driver. One would have been motivated to make such a modification to Panasyuk and Spencer so that the server and client computers both share the same image as a result of transmitting the captured GDI requests from the device driver as taught by Schneider et al.

Panasyuk discloses claim 3 in that the server and client node communicate via one of a list of industry-standard protocols. Column 3, lines 1 – 9, states, "Client nodes may communicate with server node 20 via any of a number of industry-standard data communications protocols including, but not limited to, TCP/IP, IPX/SPX, NetBEUI, or serial protocols. Alternatively, client nodes 10 may connect to server node 20 using a proprietary data communications protocol such as the ICA protocol manufactured by Citrix Systems, Inc. of Fort Lauderdale, Fla. or the RDP protocol manufactured by Microsoft Corporation of Redmond, Wash." Therefore, the region and graphics data to

be sent from the server to the client is gathered and stored in a format of a remoting protocol.

Panasyuk and Spencer disclose the method of claim 6 except wherein the display driver synchronously gathers graphics data by gathering drawing commands issued to a graphics device interface subsystem of an operating system of the server. Schneider teaches of gathering GDI drawing commands for transmission to a client from a server. Column 3, lines 29 – 32, states, "In general, the system of the present invention transmits a GDI representation of digitized video signals as well as mouse and keyboard signals over a communications link." Column 6, line 67, and column 7, lines 1 – 7, state, "The digitizer control application 220 then draws the whole captured screen to its local screen using Windows GDI calls. The remote control software application 200 captures those GDI requests and retransmits them to the controlling computer 12. The client software on the controlling computer 12 then re-executes the commands so that the screen of the controller 50 and the screen of the controlling computer 12 show the same image." Thus, the graphics data is gathered by collecting drawing commands issued to a graphics device interface subsystem. It would have been obvious to one having ordinary skill in the art at the time the invention was made to further modify the method of Panasyuk in view of Panasyuk and Spencer to include the method of Schneider such that the graphics data is gathered by collecting drawing commands issued to a graphics device interface subsystem. One would have been motivated to make such a modification to the invention of Panasyuk in view of Spencer so that upon sending the graphics data to a client computer, only the drawing commands are sent to

the client instead of sending bitmap images. This reduces the amount of information to be communicated and thus reduces the amount of bandwidth needed to transmit screen data from the server to a client.

Claim 15 is rejected under 35 USC 103(a) as unpatentable over Panasyuk in view of Spencer as applied to claim 1, and further in view of Sutou et al (US PGPub 2002/0035627 A1).

Panasyuk and Spencer as applied to claim 1 disclose the system of claim 15 except wherein a display driver collects the synchronously gathered region and graphics data region and a region and graphics gathering module gathers region and graphics data. The system of Sutou provides a means for remote controlling a terminal, where the display driver contains hooks that are used to capture the drawing data corresponding to the full window [0061].

It would have been obvious to one having ordinary skill in the art at the time the invention was made to further modify the invention of Panasyuk to include a device driver and control application collect the region data and graphics data and a remote control software application to capture drawing data so as to synchronously gather region and graphics data for a display, as with Sutou. One would have been motivated to make such a modification so that graphics data being sent from a server to a client would be more efficient and reduce the amount of data to be transmitted, thereby to make a prompt response to the display 2B of the control terminal 100B even in the use

of a communication line having a relatively low transmission rate – Sutou [0061]. Additionally, gathering region and graphics data in the same control software application module allows for synchronicity to be achieved between the data since they are both gathered from the same captured screen in the application.

Claims 16, 17, and 18 are rejected under 35 U.S.C. 103(a) as being unpatentable over Panasyuk in view of Spencer and Sutou as applied to claim 15 above, and further in view of Eagen.

Panasyuk in view of Spencer and Sutou disclose the engine in claims 16 and 18 except wherein a data output scheduler is associated with the display driver to send the region and graphics data to a client in a sequence and comprising a data gathering scheduler to schedule synchronous gathering a region and graphics data. Panasyuk teaches of issuing commands periodically at a predetermined rate to determine when changes to the server's desktop have occurred. Column 4, lines 3 – 8, states, "The agents 30, 40 can issue the Enum Windows command every 50 milliseconds, every 100 milliseconds, every 500 milliseconds, or at any period that allows the agent 30, 40 to rapidly determine when changes to its associated desktop environment have occurred without putting a significant computational burden on the node." The apparatus of Eagen includes presenting and removing of windows from a host terminal to a workstation. Column 8 discloses a display data manager that constructs a data stream according to a given format when information is to be displayed at a remote terminal. Lines 25 – 32 state, "When an applications program needs to communicate with a

remote terminal it calls up an applications program interface routine, one form of which is identified as a "display data manager." When information is to be displayed at a remote terminal, the display data manager constructs a data stream according to a particular format, and transmits this data stream to a workstation controller." Thus, the display data manager performs the tasks of output scheduler and gathering scheduler by constructing a data stream according to a particular format to transmit data to a workstation controller. It would have been obvious to one having ordinary skill in the art at the time the invention was made to further modify the invention of Panasyuk in view of Spencer and Sutou to include a display data manager to perform the tasks of an output scheduler and a data gathering scheduler as taught by Eagen. One would have been motivated to make such a modification so that the gathering and sending of the graphics and region data is performed periodically at a predetermined rate to coincide with the periodic updates of the server's desktop environment in Panasyuk.

Panasyuk in view of Spencer and Sutou disclose the engine in claim 17 except wherein a bandwidth compensator maintains security with respect to the synchronized region and graphics data during a condition of low bandwidth. Eagen discloses a communications device for transmitting data to a target device. Column 4, lines 65 – 67, and column 5, line 1, state, "The controlling computer 12 also includes a communications device 53 for communicating with the target device(s). Such a device 53 may include (1) a modem for connecting via a telephone connection, (2) a wireless transceiver for wirelessly communicating, and (3) a wired adapter (e.g., an Ethernet or token ring adapter)." Column 3, lines 56 – 67, describes the contents of the controlling

computer that contains the communications device as containing a CPU. It is obvious to one having ordinary skill in the art that a CPU and communications device can be combined to create a bandwidth compensator such that the system is aware of conditions of low bandwidth. It would have been obvious to one having ordinary skill in the art at the time the invention was made to further modify the invention of Panasyuk in view of Spencer and Sutou to include a bandwidth compensator. One would have been motivated to make such a modification so that in conditions of low bandwidth, the security with respect to the synchronized region and graphics data is maintained and no unintended regions of the graphics data will be displayed on the target computer.

Claims 4 and 22-24 are rejected under 35 U.S.C. 103(a) as unpatentable over Panasyuk in view of Spencer as applied to claim 1, and further in view of Grossman.

As to claim 4,

Panasyuk discloses the method of claim 4 except wherein the region data is synchronously gathered by a display driver-level window object created to contain the shape and position information. Grossman teaches of using a display driver-level window object to gather and contain the shape and position information of a shared object in figure 4. It would have been obvious to one having ordinary skill in the art at the time the invention was made to further modify the invention of Panasyuk so that the region data is synchronously gathered by a display driver-level window object to contain the shape and position information. One would have been motivated to make such a modification to the invention of Panasyuk so that there existed a record containing all

the region data of the shared window between the server and client, thus facilitating the synchronicity between the region data and the graphics data upon sending the shared window to a client by sending the region data in advance of the graphics data in the data structure.

As to claim 24,

Panasyuk describes a system and method in which a window being shown on a server display is sent and displayed on a client display. Region data describing the window region on a server desktop is gathered and sent along with graphics data for the region to a client display. Column 2, lines 31 – 42, states, "In a further aspect, the present invention relates to a system for incorporating windows from a remote desktop into a local desktop. The system comprises a local node and a remote node connected by a communications link. The communications link includes a first virtual channel and a second virtual channel. The nodes exchange desktop information such as window position, window size, and Bordering of desktop windows, over the first virtual channel. The nodes exchange graphical information over the second virtual channel. In some embodiments, the first virtual channel and the second channel may be provided as a single virtual channel." Column 6, lines 58 – 67, states, "During a seamless windowing mode session, the server agent 30 will send window information such as window position, size, styles, window text, etc. for all top-level windows on the server node. Also, foreground window information is sent, i.e., which window on the server node desktop is the foreground window. In accordance with this information, the client agent 40 creates windows with the same size/position as the server node windows on the

client node desktop. In some embodiments, window elements are transmitted as bitmaps from the server node 20." Thus, the region and graphics data are gathered to describe the shared window and then sent to a client. Column 10 describes computing device readable media containing programs to perform the invention. Panasyuk does not disclose synchronously gathering the region and graphics data and sending the data to a client while maintaining synchronicity between the region and graphics data. The invention of Grossman describes the movement of windows or icons in a transport region from one monitor to another. Figure 4 shows a data structure that defines the transport region and the destination of the transported icon or window. As shown in the data structure, the coordinate values of the transport region and the coordinate values for the target position are synchronously sent with the graphics data as represented by the icon identification number and class identification number. Column 4, lines 66 – 67, states, "Also, each graphical image within a class is uniquely identified by its icon identification number 440." Column 6, lines 6 – 10, states, "The "icon" to be transported need not be static but may consist of animated images or TV broadcasts/signals displayed in a window or icon. The target monitors may be local (e.g., on the same desktop) or in a remote location connected via a network." The icon identification number represents graphical data in that it identifies the graphical image being sent. Thus, the region data is synchronously gathered with the graphics data and the two are sent to a client from the server while maintaining the synchronicity between the two. It would have been obvious to one having ordinary skill in the art at the time the invention was made to modify the invention of Panasyuk to include synchronously gathering and

sending the region and graphics data for displaying a region of a server desktop remotely on a client display as taught by Grossman et al. One would have been motivated to make such a modification to the invention of Panasyuk with Grossman et al. so that while sharing a window between the server and client displays, the graphics being shared will always correspond to the intended region and will not display graphics data not intended to be sent to the client.

Claim 24 is disclosed by the invention of Panasyuk and Spencer in view of Grossman as above. The condition wherein the bandwidth is sufficient for sending the region and graphics data to the client was chosen from the claim, therefore placing no restrictions on the methods described by Panasyuk and Grossman. Panasyuk as modified by Grossman sends the region in synchronicity with the graphics data to the client with the region data preceding the graphics data.

Panasyuk implicitly teaches the synchronous limitation as noted above. Panasyuk in fact teaches such limitations (see for example 4:1-4:12), where it teaches that the local agents issue the Enum Windows command every time period, e.g. "The agents 30, 40 can issue the Enum Windows command every 50 milliseconds, every 100 milliseconds, every 500 milliseconds, or at any period that allows that allows the agent 30, 40 to rapidly determine when changes to its associated desktop environment have occurred without putting a significant computation burden on the node." The Enum Windows command clearly serves to determine changes in the details of the operating system under a Windows™ operating system environment. Clearly, a window defined

in the specification and as well known in the art constitutes a 'region of a server desktop', and applicant has not contested this point.

Spencer clearly teaches synchronization verification of multiple applications across remote systems, where at least one local application window is synchronized with the at least one remote application window (Abstract, 2:45-3:7, Figs. 7C and 7G). Specifically, the system provides automated real-time feedback as to the synchronization of all windows and thusly allows synchronization between them (4:59-5:27).

Clearly, it would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the system of Panasyuk with Spencer because Spencer provides methods for synchronizing applications and ensuring that such synchronization is maintained in real time in a low bandwidth and transparent manner (4:59-5:27), where clearly it provides improvement over prior art window sharing (7:44-50). Additionally, the entirety of the rejection to claim 1 is incorporated by reference. Motivation for combination with Grossman was discussed above.

As to claims 22 and 23,

The invention of Grossman describes the movement of windows or icons in a transport region from one monitor to another. Figure 4 shows a data structure that defines the transport region and the destination of the transported icon or window. As shown in the data structure, the coordinate values of the transport region and the coordinate values for the target position are synchronously sent with the graphics data as represented by the icon identification number and class identification number.

Column 4, lines 66 – 67, states, "Also, each graphical image within a class is uniquely identified by its icon identification number 440." Column 6, lines 6 – 10, states, "The "icon" to be transported need not be static but may consist of animated images or TV broadcasts/signals displayed in a window or icon. The target monitors may be local (e.g., on the same desktop) or in a remote location connected via a network." The icon identification number represents graphical data in that it identifies the graphical image being sent. Thus, the region data is synchronously gathered with the graphics data in that the two are obtained for transmission once they are moved into a designated region on the display. Therefore, the region data precedes the graphics data in the data stream structure.

Panasyuk and Spencer disclose claim 22 as the data is streamed between computers (a client and a server, a remote terminal and a local terminal). Obviously, as noted in the rejection to claim 1, which is incorporated by reference, the system of Panasyuk transmits window position, size, and graphics data in the window, which clearly constitute 'geometry of a visual region to be remotely displayed' as defined in applicant's specification and in claim 1. Clearly, as noted therein, the system of Panasyuk sends graphic data at regular intervals, as set forth with the Enum Windows command, where the region data would occur in every regular time interval with the update. Motivation and combination are taken from the rejection of claim 24 above.

As to claim 23, Spencer clearly states that synchronicity is maintained, as does Grossman, and this limitation is covered in the material incorporated by reference.

Conclusion

Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Eric Woods whose telephone number is 571-272-7775. The examiner can normally be reached on M-F 7:30-5:00.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Ulka Chauhan can be reached on 571-272-7782. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Eric Woods

June 12 2006

ULKA CHAUHAN
SUPERVISORY PATENT EXAMINER